

# BlueTACK

*Get hooked without strings*

The Bluetooth\* Protocol Stack

---

## Reference Design Application Note

Ver 1.0

**DCM**  
TECHNOLOGIES

**SEI CMM 5**

\*Bluetooth is a trademark owned by its proprietor and is used by DCM Technologies under license

© DCM Technologies

All rights reserved.

DCM Technologies reserves the right to make changes in its product without notice in order to improve design or performance characteristics.

This publication neither states nor implies any warranty of any kind.

The information is provided “as is” without any express or implied warranty of any kind including warranties of merchantability, non-infringement of intellectual property, or fitness for any particular purpose. In no event shall DCM Technologies or its suppliers be liable for any damages whatsoever arising out of the use of or inability to use the materials. DCM Technologies and its supplies further do not warrant the accuracy of completeness of the information, text, graphics or other items contained within these materials, DCM Technologies may make changes to these materials, or to the products described therein, at any time without notice.

© Bluetooth is a trademark owned by its proprietor and is used by DCM Technologies under license.

## Table of Contents

1.	INTRODUCTION .....	4
2.	PRODUCT ARCHITECTURE .....	5
	OBEX (OBject EXchange Protocol).....	6
	Routing Unit .....	6
	RFCOMM.....	6
	SDP (Service Discovery Protocol).....	6
	L2CAP (Logical Link Control and Adaptation Protocol).....	6
	Security Manager.....	6
	HCI.....	7
	Configuration Manager.....	7
	OSL (Operating System Support Library) .....	7
3.	TRANSPORT INTERFACES: .....	7
4.	MINIMUM SYSTEM REQUIREMENTS.....	7
5.	DEVICE DRIVERS FOR OPERATING SYSTEMS.....	7
6.	APPLICATION PROGRAMMING INTERFACE .....	7
	IOCTLS .....	8
	OBEX API's:.....	8
	Salutation APIs: .....	9
7.	BLUETOOTH DEVICES SUPPORT .....	9
8.	APPLICATIONS .....	9
	File Transfer Application.....	9
9.	INTEGRATING APPLICATIONS WITH BlueTACK .....	11
10.	TEST AND DEBUG PLATFORM .....	12
	TCI Application for testing L2CAP:.....	12
	Architecture of L2CAP Test Application .....	13
11.	BUILT-IN PROTOCOL ANALYZER.....	14
12.	DELIVERABLES.....	15
13.	CONTACT.....	15

## 1. INTRODUCTION

“BlueTACK” - The Bluetooth® protocol stack from DCM Technologies is a high-performance, interoperable and versatile IP solution. It broadens the scope of wireless connectivity to all digital devices, enabling them to interact seamlessly.

DCM Technologies “BlueTACK” is an easy to integrate solution, which can conveniently be adapted. It is designed for use in both embedded and non-embedded applications. It uses modular and portable software architecture.

### Salient Points

- Portability - For maximum portability the stack is implemented in ANSI C.
- OS independent Code - An operating system support library encapsulates OS-specific functionality.
- Configurability - Modular stack allows selective functionality for specific profiles/applications.
- Built in protocol analyzer-The development and debugging tool for analysis of packets. It can dynamically be switched-off or on.
- Bluetooth specification 1.0B compliant.
- All upgrades including ver1.1 will be available.
- Compact Code.
- BQB compliant and interoperability tested at Unplug Fests.

### Features

- Basic Protocol Layers
  - RFCOMM
  - L2CAP
  - SDP
  - HCI
  - Security & Configuration Management
- Operating System
  - Windows98/2000
  - Windows CE
  - VxWorks
  - MicroITRON
  - Nucleus OS
- Profiles
  - OBEX
  - Synchronization
  - LAN Access
  - Serial Port
  - General Access
  - TCS
  - Dial-up Networking

- AT Commands
- Fax / Modem
- Headset
- API
  - IOCTL
  - Salutation Mapped
  - Socket Interface

## 2. PRODUCT ARCHITECTURE

DCM BlueTACK-The Bluetooth Protocol Stack is Operating System Independent. The Virtual Operating System concept makes BlueTACK directly adaptable to a wide range of RTOS (Real Time Operating Systems) for embedded systems like WinCE, VxWorks, Nucleus, microITRON, EPOC, pSOS etc., as well as standard operating systems such as Windows and UNIX\*). BlueTACK can also be ported to more basic operating systems (e.g. single thread system) without changing the source code for the protocol layers.

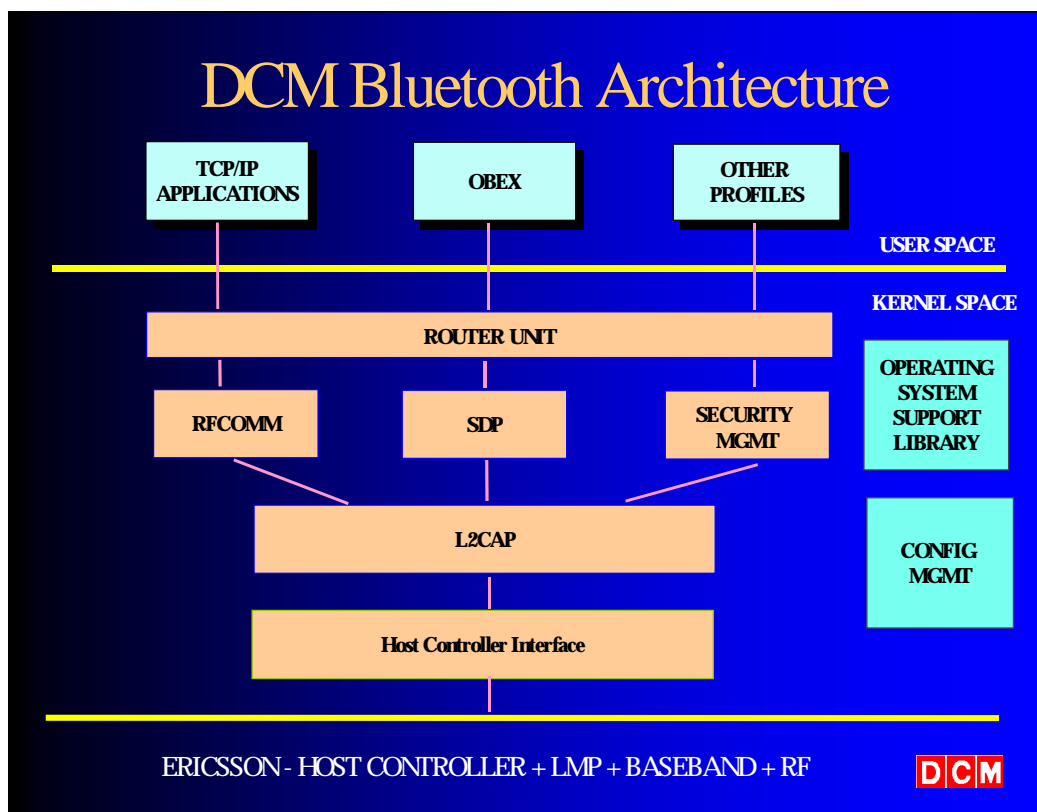


Figure 1: DCM BlueTACK Architecture

## **OBEX (OBject EXchange Protocol)**

OBEX is a session protocol developed by the Infrared Data Association (IrDA) to exchange objects in a simple and spontaneous manner. OBEX provides the same basic functionality as HTTP but in a much lighter fashion, uses a client-server model and is independent of the transport mechanism and transport API (Application Programming Interface), provided it realizes a reliable transport base.

## **Routing Unit**

This module acts as a routing Unit between Applications and Protocol layer. It looks at the packet and forwards it appropriately to RFCOMM, SDP, L2CAP and Security Manager and vice versa.

## **RFCOMM**

RFCOMM is a serial line emulation protocol and is based on ETSI 07.10 specification. This “cable replacement” protocol emulates RS-232 control and data signals over Bluetooth base-band, providing both transport capabilities for upper level services (e.g. OBEX) that use serial line as transport mechanism.

## **SDP (Service Discovery Protocol)**

Discovery services are crucial part of the Bluetooth framework. These services provide the basis for all the usage models. Using SDP, device information, services and the characteristics of the services can be queried and after that, a connection between two or more Bluetooth devices can be established.

## **L2CAP (Logical Link Control and Adaptation Protocol)**

L2CAP provides Channel Management services to the upper layer like RFCOMM, SDP. It enables them to transmit and receive L2CAP data packets. The maximum size of data packets can be 64 kilobytes.

This module has a sub-module - Connection Manager for controlling the overall states and modes of the various connections present in the Bluetooth device. It receives all the connection related callbacks and maintains the connection database. This way it also multiplexes ACL (Asynchronous Connection Less) and SCO (Synchronous Connection Oriented) links into a common control.

## **Security Manager**

This module controls the security features that are used by different applications and Multiplexing protocols. These features include Authentication (Unidirectional or mutual) and Encryption.

## **HCI**

Host Controller Interface is the interface between the protocol stack and Bluetooth module. It provides functionality of base-band and LMP (Link Manager Protocol) to the Bluetooth protocol stack. This layer keeps the Bluetooth protocol stack independent of specific Bluetooth module.

## **Configuration Manager**

This module controls the various configuration items like Country Code, etc.

## **OSL (Operating System Support Library)**

The objective of this module is to keep all OS dependent calls in a separate module to increase the portability of the protocol stack. Whenever, the protocol is to be ported to a new operating system, this module is redeveloped mapping all calls to the OS specific calls.

## **3. TRANSPORT INTERFACES:**

The architecture of DCM BlueTACK is compatible with various transport interfaces to the Bluetooth Base-band module. Its variants include BlueTACK USB, BlueTACK UART as well as BlueTACK PCMCIA card.

## **4. MINIMUM SYSTEM REQUIREMENTS**

Minimum system requirements for using the BlueTACK are as follows:

1. Pentium processor
2. 64 MB RAM

## **5. DEVICE DRIVERS FOR OPERATING SYSTEMS**

DCM BlueTACK has following ready driver choice

- Windows 2000/NT
- VxWorks (RTOS)
- Windows CE

## **6. APPLICATION PROGRAMMING INTERFACE**

DCM BlueTACK comes with an extensive APIs for developing powerful Bluetooth applications. DCM BlueTACK hosts profile specific APIs as well as APIs to access all layers of the protocol stack. The APIs come in two variants - Blocking and Non-Blocking calls. Application designers have the flexibility to use any one of the call mechanisms that best suits their application.

APIs supporting above profiles are in the form of IOCTL's. For different modules of the Bluetooth stack, separate IOCTLs have been exported.

## IOCTLS

The following IOCTLs are available to user for accessing the functionalities of various protocol layers:

- IOCTL\_RFCOMM
- IOCTL\_CONFIGURATION
- IOCTL\_LISTEN
- IOCTL\_SERV\_DISC
- IOCTL\_SDP
- IOCTL\_L2CAP
- IOCTL\_PROTOCOL\_ANALYZER

## OBEX API's:

Following APIs are available for developing applications on top of OBEX:

- **OBJEX\_OpenSession:**
  - API for client application to open an OBEX session with OBEX Server.
- **OBJEX\_CloseSession:**
  - API for client application to close an OBEX session.
- **OBJEX\_StartServer:**
  - Services can use this function to start an OBEX server that gives events corresponding to the requests received.
- **OBJEX\_StopServer:**
  - This API is for stopping the server.
- **OBJEX\_AbortOperation:**
  - This is to abort the current operation. This function can be called any time whenever some Object transfer operation is on going that needs to be stopped immediately.
- **OBJEX\_PutObject:**
  - For PUSHING an object to the server. Object information is passed with the data.
- **OBJEX\_GetObject:**
  - For PULLING an object from the server. Object information is obtained with the data.
- **OBJEX\_SetPath:**
  - This API is to set the current working folder and to create a new folder at the server.



### Salutation APIs:

Following APIs are available:

- **slmRegisterCapability**
  - To register the capability/service.
- **slmUnRegisterCapability**
  - To un-register a capability/service.
- **slmSearchCapability**
  - To search the capability/service.
- **slmQueryCapability**
  - To discover registered functional units and their capabilities.

## 7. BLUETOOTH DEVICES SUPPORT

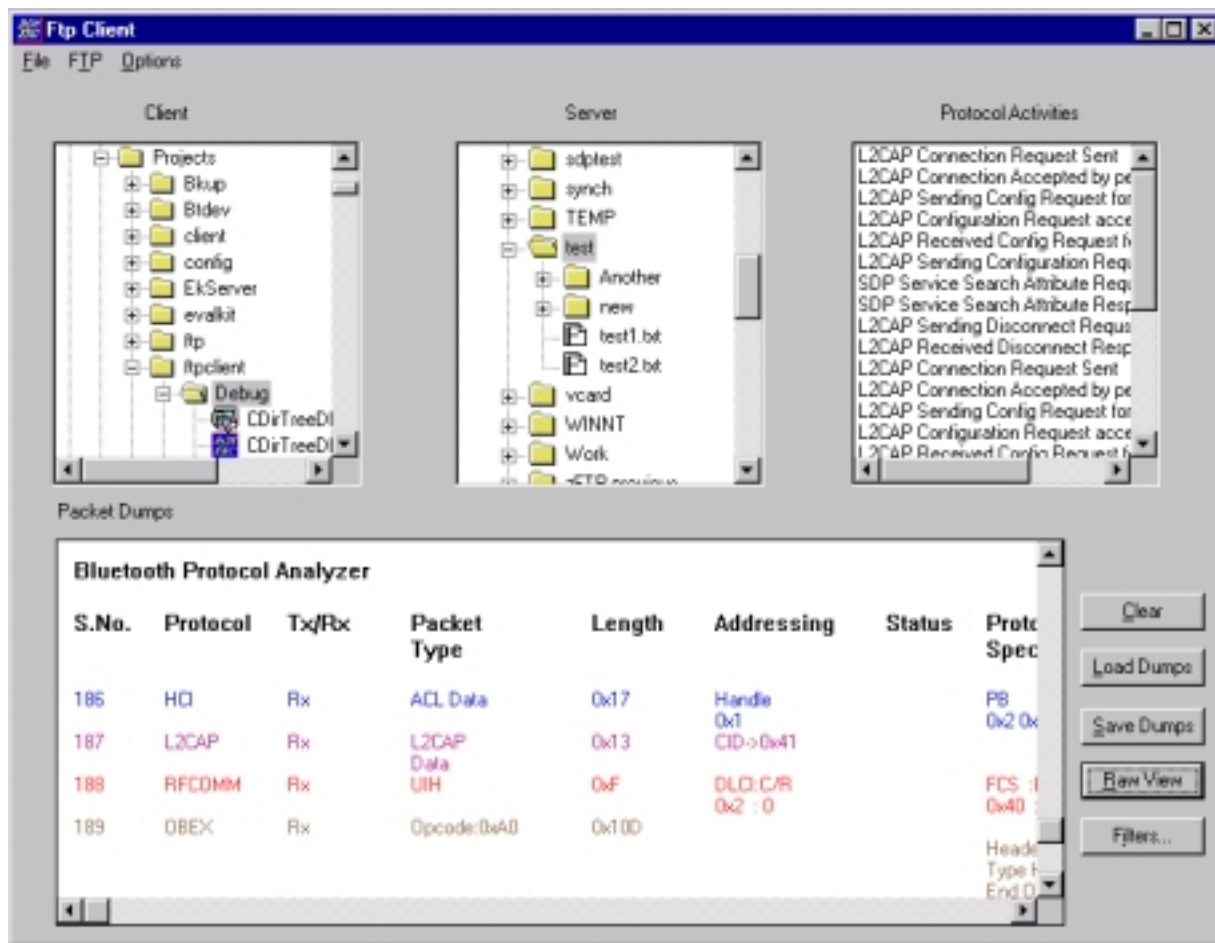
The BlueTACK has flexibility to integrate with any hardware module, which is compatible with standard HCI interface. This allows OEM's the freedom to chose the base-band hardware of their choice. The BlueTACK is currently tested on Ericsson BT base-band and RF module.

## 8. APPLICATIONS

DCM BlueTACK comes with ready File Transfer Application for windows and DUN for Internet Browsing.

### File Transfer Application

This application demonstrates the file transfer on Client/Server architecture through Bluetooth wireless devices. Similar to the Windows Explorer, the client as well as the server can be browsed in the form of tree view and the desired operation through Copy-Paste can be performed. A user-friendly Graphical User Interface is provided. Following is a typical view of FTP client application.



The application window contains a menu bar, four windows and several buttons. The functioning of all the above mentioned controls is described below:

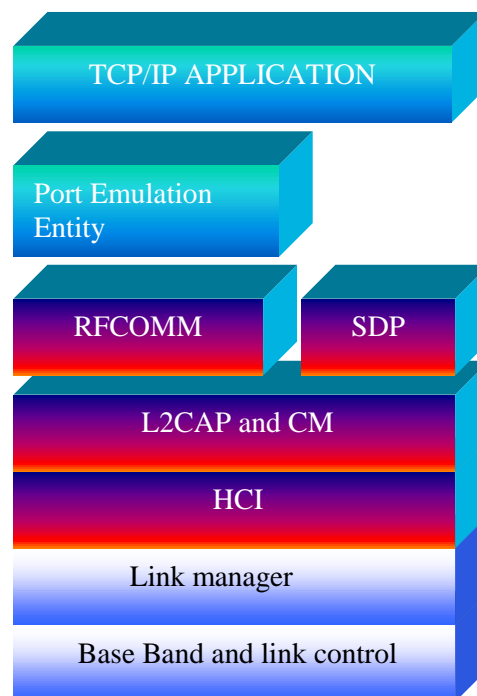
- ❖ The menu bar has following menu options:
  - File.
    - ❖ Exit
  - FTP.
    - ❖ Copy
    - ❖ Paste
    - ❖ Create
    - ❖ Delete
  - Options
    - ❖ Browse devices
    - ❖ Disconnect
    - ❖ Show protocol Analyzer
    - ❖ Show protocol Activities
- ❖ “Client” window displays the contents of folders on Client side. By default the “current folder” on client is set as the path from which the client application is executed.

- ❖ “Server” window displays the folder listing of root folder of Server side. By default the “Root folder” is set as current folder on server.
- ❖ The application contains a “Protocol Activities” window showing the list of protocol activities performed on every operation.
- ❖ The application contains a Bluetooth Protocol Analyzer window that displays the details of transmit and receive activities. Bluetooth Protocol Analyzer shows the packets being transmitted or received corresponding to the commands. It is explained in section 11.

## 9. INTEGRATING APPLICATIONS WITH BlueTACK

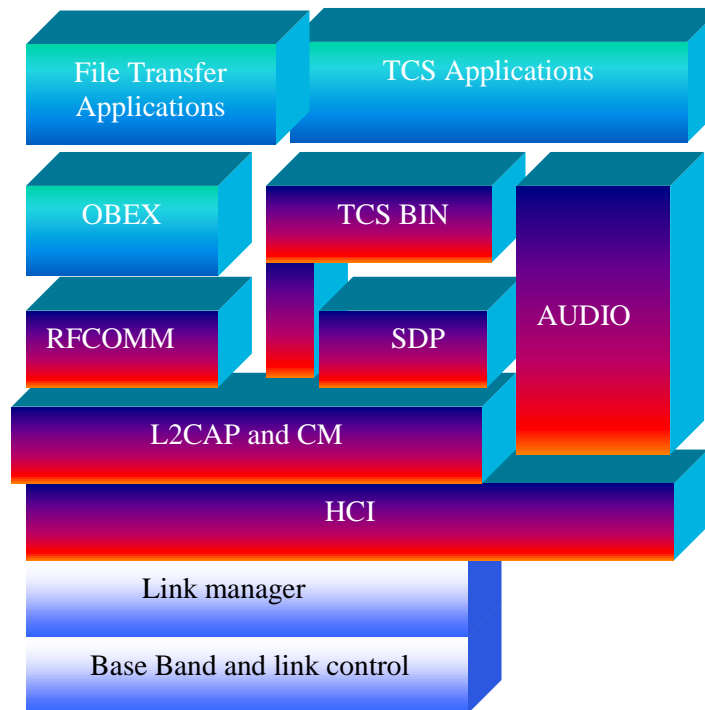
The following example discusses how an application that requires a serial communications link for the transfer of data can be added to BlueTACK. RFCOMM layer provides the serial communication emulation.

RFCOMM requires a Port Emulation Entity (PEE) to communicate with applications. The PEE provides the adaptation between the stack and the application environment. The assumption here is that the application is using Bluetooth as a simple communications media, like a wireless replacement for a serial cable.



**Figure 2a: Showing the integrated application with stack.**

Following model describes the OBJECT EXCHANGE APPLICATIONS such as File Transfer and applications over TCS:



**Figure 2b: Showing the integrated application with stack.**

## 10. TEST AND DEBUG PLATFORM

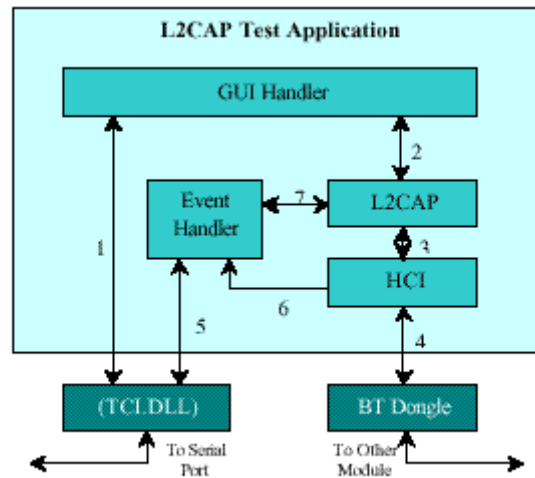
### Test Control Interface (TCI ) Application for testing L2CAP:

The Bluetooth Test Control Interface, TCI, developed by DCM as per the specification is used for verifying the implemented functionality of the Logical Link Control and Adaptation Protocol, L2CAP of Bluetooth Stack.

The TCI - L2CAP interface is based on the HCI and is used during verification of the L2CAP layer of the SUT/IUT. It uses the general event and command syntax as per specification. Commands and events are defined according to the specified L2CAP service interface.

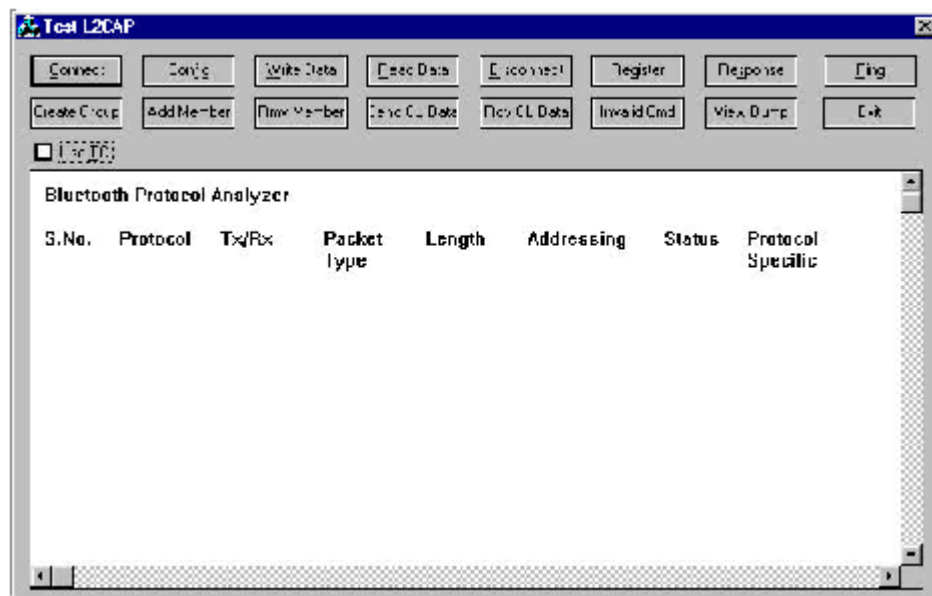
## Architecture of L2CAP Test Application

The following block diagram can illustrate the architecture of the application.



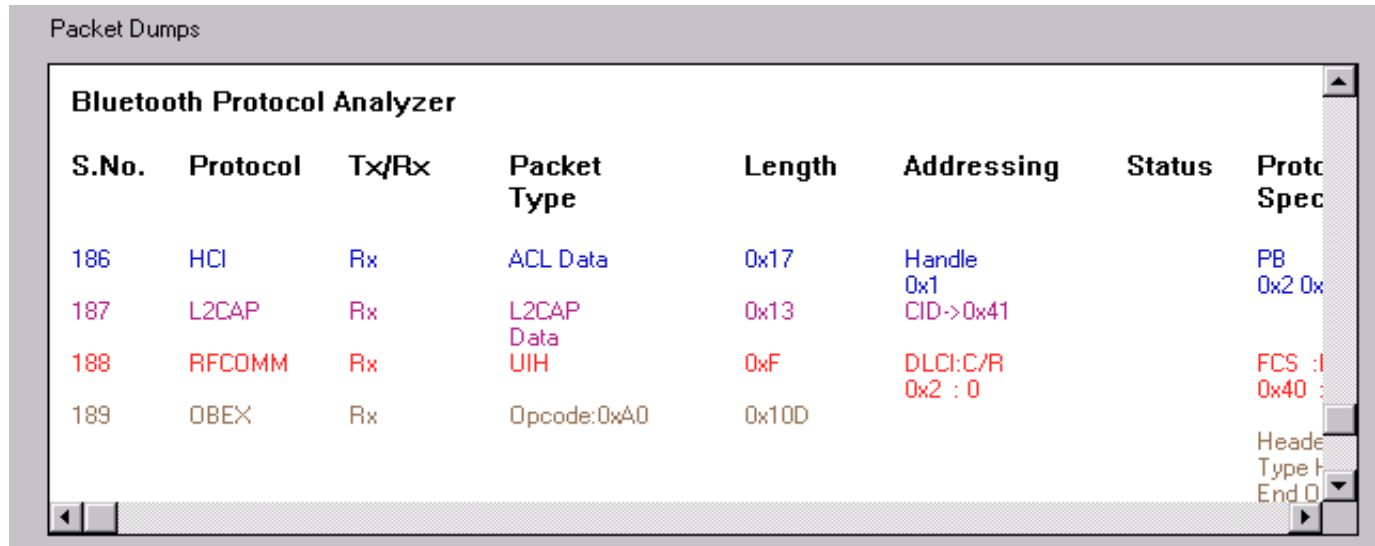
**Figure3: showing the architecture of L2CAP test application**

The following window shows the TCI application. It contains the buttons corresponding to the L2CAP commands.



## 11. BUILT-IN PROTOCOL ANALYZER

The DCM BlueTACK has built in Bluetooth Protocol Analyzer. It keeps track of all the protocol activities at every layer and displays them in the Analyzer window. This unique feature is a handy tool for application developers to monitor data and control flow inside the protocol stack. Figure below shows a snapshot of protocol activities as captured by Protocol Analyzer.



The screenshot shows a window titled "Packet Dumps" containing a sub-window titled "Bluetooth Protocol Analyzer". The sub-window displays a table with the following columns: S.No., Protocol, Tx/Rx, Packet Type, Length, Addressing, Status, and Protocol Specific. The table contains four rows of data, each representing a captured packet.

S.No.	Protocol	Tx/Rx	Packet Type	Length	Addressing	Status	Protocol Specific
186	HCI	Rx	ACL Data	0x17	Handle 0x1		PB 0x2 0x
187	L2CAP	Rx	L2CAP Data	0x13	CID->0x41		
188	RFCOMM	Rx	UIH	0xF	DLCI:C/R 0x2 : 0		FCS : 0x40 :
189	OBEX	Rx	Opcode:0xA0	0x10D			Heade Type h End O

Various columns in the Bluetooth Protocol Analyzer window are as follows:

1. **S. No.:** Packet sequence number to view the raw dump of the packet.
2. **Protocol:** The name of the protocol layer to which the packet belongs.
3. **Tx/Rx:** Indicates that the packet is transmitted or received.
4. **Packet Type:** Protocol specific packet type.
5. **Length:** Length field inside the packet.
6. **Addressing:** Destination related information, like Channel Id, DLCI etc.
7. **Status:** If packet contains some specific status info, it is shown here.
8. **Protocol Specific:** Some other protocol/packet related important parameter values.

## **12. DELIVERABLES**

The DCM BlueTACK has the following deliverables:

- ANSI C source code for various layers and profiles
- User Manual
- Sample Application
- Reference design and Porting Guidelines